

1 1. (Original) An apparatus comprising:  
2 at least one processor;  
3 a memory coupled to the at least one processor;  
4 a cluster engine residing in the memory and executed by the at least one  
5 processor;  
6 a job residing in the memory and executed by the at least one processor, the job  
7 including:  
8 at least one work thread that performs at least one predefined task; and  
9 a main thread that receives messages from at least one computer system  
10 coupled to the apparatus, that routes appropriate messages from the at least one  
11 computer system to the at least one work thread, and that signals to the cluster  
12 engine when at least one fault occurs when the at least one work thread performs  
13 the at least one predefined task.

1 2. (Original) The apparatus of claim 1 wherein the at least one predefined task comprises  
2 a protocol that includes at least one acknowledge (ACK) round, and that performs only  
3 local processing between ACK rounds.

1 3. (Original) The apparatus of claim 1 wherein the main thread performs only local  
2 processing.

1 4. (Original) The apparatus of claim 1 wherein the main thread does not wait for any  
2 local resource, and thus is guaranteed to receive a message sent by the cluster engine.

1 5. (Original) The apparatus of claim 1 wherein the signal to the cluster engine comprises  
2 an unregistration with the cluster engine

1 6. (Original) The apparatus of claim 5 wherein the unregistration with the cluster engine  
2 causes the cluster engine to generate a membership change message.

1 7. (Currently Amended) A networked computer system comprising:  
2 a cluster of computer systems that each includes:  
3 a network interface that couples each computer system via a network to  
4 other computer systems in the cluster;  
5 at least one processor;  
6 a memory coupled to the at least one processor;  
7 a cluster engine residing in the memory and executed by the at least one  
8 processor;  
9 a job residing in the memory and executed by the at least one processor,  
10 the job including:  
11 at least one work thread that executes a predefined protocol that  
12 includes at least one acknowledge (ACK) round, wherein the protocol only  
13 performs local tasks between ACK rounds; and  
14 a main thread that registers with the cluster engine to become a  
15 member of a group, that receives messages from at least one computer  
16 system coupled to the apparatus, that routes appropriate messages from the  
17 at least one computer system to the at least one work thread, and that  
18 signals to the cluster engine when at least one fault occurs when the at  
19 least one work thread performs the at least one predefined task by  
20 unregistering with the cluster engine, wherein unregistering with the  
21 cluster engine causes the cluster engine to generate a membership change  
22 to remaining members of the group.

1 8. (Original) A computer-implemented method for notifying jobs that form a group in a  
2 clustered computing environment when a member of the group is no longer alive, the  
3 method comprising the steps of:  
4       defining a task;  
5       providing a cluster engine for each member of the group that communicates with  
6 the other cluster engines in the group;  
7       providing at least one work thread for each job that executes the task;  
8       providing a main thread for each job, the main thread performing the steps of:  
9           receiving messages from other members of the group via the cluster engine  
10          corresponding to the main thread;  
11          routing appropriate messages from the other members of the group to the  
12          at least one work thread; and  
13          signaling to the cluster engine when at least one fault occurs during the  
14          execution of the task by the work thread.

1 9. (Original) The method of claim 8 wherein the task comprises a protocol that includes  
2 at least one acknowledge (ACK) round, and that performs only local processing between  
3 ACK rounds.

1 10. (Original) The method of claim 8 wherein the main thread performs only local  
2 processing.

1 11. (Original) The method of claim 8 wherein the main thread does not wait for any local  
2 resource, and thus is guaranteed to receive a message sent by the cluster engine.

1 12. (Original) The method of claim 8 wherein the step of signaling to the cluster engine  
2 comprises the step of unregistering with the cluster engine.

1 13. (Original) The method of claim 12 wherein the step of unregistering with the cluster  
2 engine causes the cluster engine to generate a membership change message to remaining  
3 members of the group.

1 14. (Original) A computer-implemented method for notifying jobs that form a group in a  
2 clustered computing environment when a member of the group is no longer alive, the  
3 method comprising the steps of:

4 defining a protocol that includes at least one acknowledge (ACK) round, and that  
5 performs only local processing between ACK rounds;

6 providing a cluster engine for each member of the group that communicates with  
7 the other cluster engines in the group;

8 providing at least one work thread for each job that executes at least a portion of  
9 the protocol;

10 providing a main thread for each job, the main thread performing the steps of:

11 registering with the cluster engine to become a member of the group;

12 receiving messages from other members of the group via the cluster engine  
13 corresponding to the main thread;

14 routing appropriate messages from the other members of the group to the  
15 at least one work thread;

16 wherein the main thread performs only local processing and does not wait  
17 for any local resource, and thus is guaranteed to receive a message sent by the  
18 cluster engine;

19 unregistering with the cluster engine when at least one fault occurs during  
20 execution of the protocol.

1 15. (Original) The method of claim 14 wherein the step of unregistering with the cluster  
2 engine causes the cluster engine to generate a membership change message to remaining  
3 members of the group.

1 16. (Original) A program product comprising:  
2 (A) a computer program comprising:  
3 at least one work thread that performs at least one predefined task; and  
4 a main thread that receives messages from a corresponding cluster engine,  
5 that routes appropriate messages from the cluster engine to the at least one work  
6 thread, and that signals to the cluster engine when at least one fault occurs when  
7 the at least one work thread performs the at least one predefined task; and  
8 (B) signal bearing media bearing the computer program.

1 17. (Original) The program product of claim 16 wherein the signal bearing media  
2 comprises recordable media.

1 18. (Original) The program product of claim 16 wherein the signal bearing media  
2 comprises transmission media.

1 19. (Original) The program product of claim 16 wherein the at least one predefined task  
2 comprises a protocol that includes at least one acknowledge (ACK) round, and that  
3 performs only local processing between ACK rounds.

1 20. (Original) The program product of claim 16 wherein the main thread performs only  
2 local processing.

1 21. (Original) The program product of claim 16 wherein the main thread does not wait  
2 for any local resource, and thus is guaranteed to receive a message sent by the cluster  
3 engine.

1 22. (Original) The program product of claim 16 wherein the signal to the cluster engine  
2 comprises an unregistration with the cluster engine

1 23. (Original) The program product of claim 22 wherein the unregistration with the  
2 cluster engine causes the cluster engine to generate a membership change message.

1 24. (Original) A program product comprising:

2 (A) a computer program comprising:

3 at least one work thread that performs a predefined protocol that  
4 includes at least one acknowledge (ACK) round, wherein the protocol only  
5 performs local tasks between ACK rounds; and

6 a main thread that registers with the cluster engine to become a  
7 member of a group, that receives messages from at least one computer  
8 system coupled to the apparatus, that routes appropriate messages from the  
9 at least one computer system to the at least one work thread, and that  
10 signals to the cluster engine when at least one fault occurs when the at  
11 least one work thread performs the at least one predefined task by  
12 unregistering with the cluster engine, wherein unregistering with the  
13 cluster engine causes the cluster engine to generate a membership change  
14 to remaining members of the group; and

15 (B) signal bearing media bearing the computer program.

1 25. (Original) The program product of claim 24 wherein the signal bearing media  
2 comprises recordable media.

1 26. (Original) The program product of claim 24 wherein the signal bearing media  
2 comprises transmission media.

---

### **STATUS OF THE CLAIMS**

Claims 1-26 were originally filed in this patent application. In the pending office action, claim 7 was rejected under 35 U.S.C. §112, second paragraph. Claims 1-26 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,108,699 to Moiin. No claim was allowed. In this amendment, claim 7 has been amended. Claims 1-26 are currently pending.